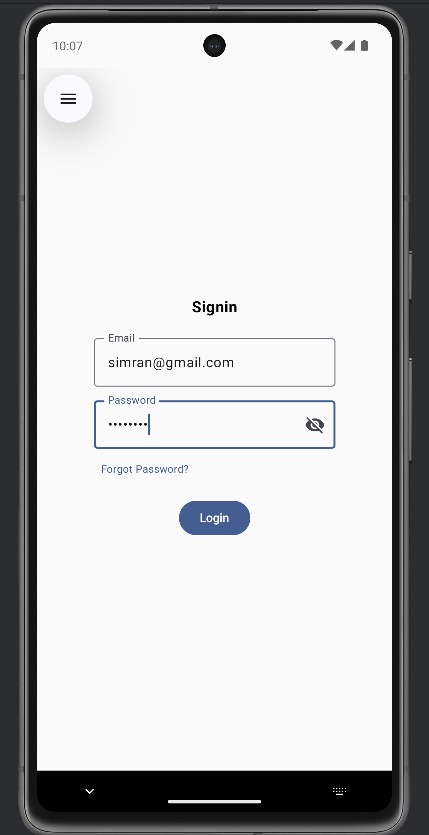
**MainActivity Code:**

package com.example.login\_page  
  
import android.os.Bundle  
import androidx.activity.ComponentActivity  
import androidx.activity.compose.setContent  
import androidx.activity.enableEdgeToEdge  
import androidx.compose.foundation.clickable  
import androidx.compose.foundation.layout.\*  
import androidx.compose.foundation.text.KeyboardOptions  
import androidx.compose.material.icons.Icons  
import androidx.compose.material.icons.filled.Visibility  
import androidx.compose.material.icons.filled.VisibilityOff  
import androidx.compose.material3.\*  
import androidx.compose.runtime.\*  
import androidx.compose.ui.Alignment  
import androidx.compose.ui.Modifier  
import androidx.compose.ui.text.font.FontWeight  
import androidx.compose.ui.text.input.KeyboardType  
import androidx.compose.ui.text.input.PasswordVisualTransformation  
import androidx.compose.ui.text.input.VisualTransformation  
import androidx.compose.ui.tooling.preview.Preview  
import androidx.compose.ui.unit.dp  
import androidx.compose.ui.unit.sp  
import androidx.navigation.NavController  
import androidx.navigation.compose.NavHost  
import androidx.navigation.compose.composable  
import androidx.navigation.compose.rememberNavController  
import com.example.Login\_Page.AppNavigation  
import com.example.login\_page.ui.theme.Login\_PageTheme  
import java.util.regex.Pattern  
  
class MainActivity : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 enableEdgeToEdge()  
 setContent {  
 Login\_PageTheme {  
 AppNavigation()  
 }  
 }  
 }  
}  
  
  
@Preview(showBackground = true)  
@Composable  
fun LoginPreview() {  
 LoginPage(rememberNavController())  
}

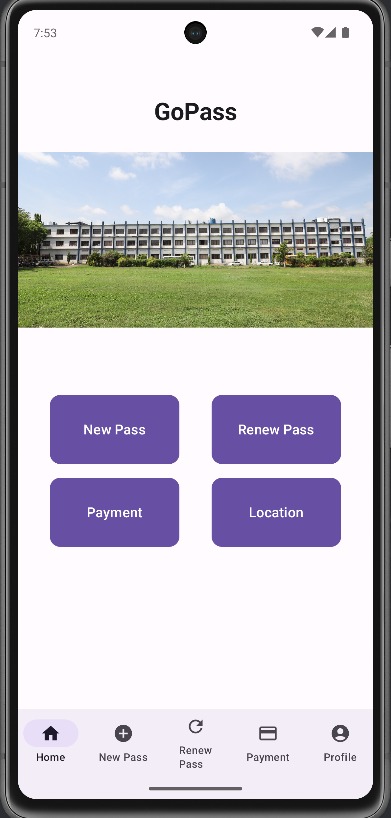
**Login Page Rough Code:**

package com.example.login\_page  
  
import android.os.Bundle  
import androidx.activity.ComponentActivity  
import androidx.activity.compose.setContent  
import androidx.activity.enableEdgeToEdge  
import androidx.compose.foundation.clickable  
import androidx.compose.foundation.layout.\*  
import androidx.compose.foundation.text.KeyboardOptions  
import androidx.compose.material.icons.Icons  
import androidx.compose.material.icons.filled.*Visibility*import androidx.compose.material.icons.filled.*VisibilityOff*import androidx.compose.material3.\*  
import androidx.compose.runtime.\*  
import androidx.compose.ui.Alignment  
import androidx.compose.ui.Modifier  
import androidx.compose.ui.text.font.FontWeight  
import androidx.compose.ui.text.input.KeyboardType  
import androidx.compose.ui.text.input.PasswordVisualTransformation  
import androidx.compose.ui.text.input.VisualTransformation  
import androidx.compose.ui.tooling.preview.Preview  
import androidx.compose.ui.unit.dp  
import androidx.compose.ui.unit.sp  
import com.example.login\_page.ui.theme.Login\_PageTheme  
  
class MainActivity : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 *enableEdgeToEdge*()  
 *setContent* **{** Login\_PageTheme **{** LoginPage()  
 **}  
 }** }  
}  
  
@Composable  
fun LoginPage() {  
 var email by remember **{** *mutableStateOf*("") **}** var password by remember **{** *mutableStateOf*("") **}** var passwordVisible by remember **{** *mutableStateOf*(false) **}** Column(  
 modifier = Modifier  
 .*fillMaxSize*()  
 .*padding*(16.*dp*),  
 verticalArrangement = Arrangement.Center,  
 horizontalAlignment = Alignment.CenterHorizontally  
 ) **{** Text(text = "Signin", fontSize = 18.*sp*, fontWeight = FontWeight.Bold)  
 Spacer(modifier = Modifier.*height*(16.*dp*))  
 OutlinedTextField(  
 value = email,  
 onValueChange = **{** email = **it }**,  
 label = **{** Text("Email") **}**,  
 keyboardOptions = KeyboardOptions(keyboardType = KeyboardType.Email)  
 )  
 Spacer(modifier = Modifier.*height*(8.*dp*))  
 PasswordField(password, passwordVisible) **{** newValue **->** password = newValue **}** Spacer(modifier = Modifier.*height*(16.*dp*))  
 Button(onClick = **{** /\* Handle login action \*/ **}**) **{** Text("Login")  
 **}  
 }**}  
  
@OptIn(ExperimentalMaterial3Api::class)  
@Composable  
fun PasswordField(  
 password: String,  
 passwordVisible: Boolean,  
 onPasswordChange: (String) -> Unit  
) {  
 var passwordVisibility by remember **{** *mutableStateOf*(passwordVisible) **}** Column **{** OutlinedTextField(  
 value = password,  
 onValueChange = onPasswordChange,  
 label = **{** Text("Password") **}**,  
 visualTransformation = if (passwordVisibility) VisualTransformation.None else PasswordVisualTransformation(),  
 keyboardOptions = KeyboardOptions(keyboardType = KeyboardType.Password),  
 trailingIcon = **{** val icon =  
 if (passwordVisibility) Icons.Filled.*Visibility* else Icons.Filled.*VisibilityOff* val description = if (passwordVisibility) "Hide password" else "Show password"  
  
 IconButton(onClick = **{** passwordVisibility = !passwordVisibility **}**) **{** Icon(imageVector = icon, contentDescription = description)  
 **}  
 }** )  
 Spacer(modifier = Modifier.*height*(8.*dp*))  
 Text(  
 text = "Forgot Password?",  
 color = MaterialTheme.colorScheme.primary,  
 style = MaterialTheme.typography.bodySmall,  
 modifier = Modifier  
 .*clickable* **{** /\* Handle forgot password \*/ **}** )  
 **}**}  
  
@Preview(showBackground = true)  
@Composable  
fun LoginPreview() {  
 Login\_PageTheme **{** LoginPage()  
 **}**}



**Home Page Rough Code:**

package com.example.rough\_code  
  
import android.os.Bundle  
import androidx.activity.ComponentActivity  
import androidx.activity.compose.setContent  
import androidx.activity.enableEdgeToEdge  
import androidx.compose.foundation.layout.fillMaxSize  
import androidx.compose.foundation.layout.padding  
import androidx.compose.material3.Scaffold  
import androidx.compose.material3.Text  
import androidx.compose.runtime.Composable  
import androidx.compose.ui.Modifier  
import androidx.compose.ui.tooling.preview.Preview  
import com.example.rough\_code.ui.theme.Rough\_CodeTheme  
import androidx.compose.foundation.Image  
import androidx.compose.foundation.layout.Arrangement  
import androidx.compose.foundation.layout.Box  
import androidx.compose.foundation.layout.Column  
import androidx.compose.foundation.layout.Row  
import androidx.compose.foundation.layout.Spacer  
import androidx.compose.foundation.layout.aspectRatio  
import androidx.compose.foundation.layout.fillMaxHeight  
import androidx.compose.foundation.layout.fillMaxSize  
import androidx.compose.foundation.layout.fillMaxWidth  
import androidx.compose.foundation.layout.height  
import androidx.compose.foundation.layout.padding  
import androidx.compose.foundation.layout.paddingFrom  
import androidx.compose.foundation.layout.size  
import androidx.compose.foundation.shape.RoundedCornerShape  
import androidx.compose.material.icons.Icons  
import androidx.compose.material3.Button  
import androidx.compose.material3.ButtonDefaults  
import androidx.compose.material3.Icon  
import androidx.compose.material3.NavigationBar  
import androidx.compose.material3.NavigationBarItem  
import androidx.compose.material3.Text  
import androidx.compose.runtime.getValue  
import androidx.compose.runtime.mutableStateOf  
import androidx.compose.runtime.remember  
import androidx.compose.runtime.setValue  
import androidx.compose.ui.Alignment  
import androidx.compose.ui.draw.clip  
import androidx.compose.ui.res.painterResource  
import androidx.compose.ui.text.font.FontWeight  
import androidx.compose.ui.unit.dp  
import androidx.compose.ui.unit.sp  
import androidx.compose.material.icons.filled.Home  
import androidx.compose.material.icons.filled.AccountCircle  
import androidx.compose.material.icons.filled.CreditCard  
import androidx.compose.material.icons.filled.Refresh  
import androidx.compose.material.icons.filled.AddCircle  
  
  
  
class MainActivity : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 enableEdgeToEdge()  
 setContent {  
 MainScreen()  
 }  
 }  
 }  
@Composable  
fun MainScreen() {  
 var selectedTab by remember { mutableStateOf(0)}  
  
 Scaffold(  
 bottomBar = {  
 BottomNavigationBar(selectedTab) { selectedTab = it }  
 }  
 ) { paddingValues ->  
 Column(  
 modifier = Modifier  
 .fillMaxSize()  
 .padding(paddingValues),  
 horizontalAlignment = Alignment.CenterHorizontally  
 ) {  
 when (selectedTab) {  
 0 -> HomePage()  
 1 -> PlaceholderScreen("Apply for New Pass")  
 2 -> PlaceholderScreen("Renew\nPass")  
 3 -> PlaceholderScreen("Payment")  
 4 -> PlaceholderScreen("Profile")  
 }  
 }  
 }  
}  
  
@Composable  
fun BottomNavigationBar(selectedTab: Int, onTabSelected: (Int) -> Unit) {  
 NavigationBar {  
 val items = listOf(  
 Pair("Home", Icons.Filled.Home),  
 Pair("New Pass", Icons.Filled.AddCircle),  
 Pair("Renew\nPass", Icons.Filled.Refresh),  
 Pair("Payment", Icons.Filled.CreditCard),  
 Pair("Profile", Icons.Filled.AccountCircle)  
 )  
  
 items.forEachIndexed { index, (title, icon) ->  
 NavigationBarItem(  
 selected = selectedTab == index,  
 onClick = { onTabSelected(index) },  
 label = { Text(title) },  
 icon = { Icon(imageVector = icon, contentDescription = title) }  
 )  
 }  
 }  
 }  
  
  
  
@Composable  
fun HomePage() {  
  
 Column(  
 modifier = Modifier  
 .fillMaxSize().padding(top = 50.dp),  
  
 horizontalAlignment = Alignment.CenterHorizontally  
  
 ) {  
 Text(text = "GoPass", fontSize = 28.sp, fontWeight = FontWeight.Bold)  
  
 Spacer(modifier = Modifier.height(16.dp))  
 Image(  
 painter = painterResource(id = R.drawable.img1),  
 contentDescription = "Rounded Image",  
 modifier = Modifier.fillMaxWidth() // Makes image take full width  
 .aspectRatio(16f / 9f)  
 .clip(RoundedCornerShape(16.dp)) // Apply rounded corners  
 )  
  
 Spacer(modifier = Modifier.height(64.dp))  
 Column(  
 modifier = Modifier.fillMaxWidth(),  
 horizontalAlignment = Alignment.CenterHorizontally,  
  
 ) {  
 Row(  
 modifier = Modifier.fillMaxWidth(),  
 horizontalArrangement = Arrangement.SpaceEvenly,  
 ) {  
 CustomButton(text = "New Pass")  
 CustomButton(text = "Renew Pass")  
 }  
  
 Spacer(modifier = Modifier.height(16.dp)) // Space between rows  
  
 Row(  
 modifier = Modifier.fillMaxWidth(),  
 horizontalArrangement = Arrangement.SpaceEvenly  
 ) {  
 CustomButton(text = "Payment")  
 CustomButton(text = "Location")  
 }  
 }  
 }  
}  
  
  
@Composable  
fun CustomButton(text: String) {  
 Button(  
 onClick = { /\* TODO: Handle Click \*/ },  
 modifier = Modifier  
 .size(width = 150.dp, height = 80.dp), // Adjust button size  
 shape = RoundedCornerShape(12.dp),  
 colors = ButtonDefaults.buttonColors()  
 ) {  
 Text(text = text, fontSize = 16.sp)  
 }  
}  
@Composable  
fun PlaceholderScreen(title: String) {  
 Box(  
 modifier = Modifier.fillMaxSize(),  
 contentAlignment = Alignment.Center  
 ) {  
 Text(text = title, fontSize = 24.sp, fontWeight = FontWeight.Bold)  
 }  
}

****

**New\_Pass Rough Code:**

package com.example.login\_page  
  
import androidx.compose.ui.platform.LocalContext  
import androidx.navigation.NavController  
  
import android.os.Bundle  
import androidx.activity.ComponentActivity  
import androidx.activity.compose.setContent  
import androidx.compose.foundation.layout.\*  
import androidx.compose.foundation.text.BasicTextField  
import androidx.compose.foundation.text.KeyboardOptions  
import androidx.compose.material3.\*  
import androidx.compose.runtime.\*  
import androidx.compose.ui.Modifier  
import androidx.compose.ui.text.input.KeyboardType  
import androidx.compose.ui.unit.dp  
import androidx.compose.ui.tooling.preview.Preview  
import androidx.navigation.compose.NavHost  
import androidx.navigation.compose.composable  
import androidx.navigation.compose.rememberNavController  
  
  
  
@Composable  
fun BusPassForm(navController: NavController) {  
  
 var name by remember { mutableStateOf("") }  
 var studentId by remember { mutableStateOf("") }  
 var contact by remember { mutableStateOf("") }  
 var email by remember { mutableStateOf("") }  
 var pickupLocation by remember { mutableStateOf("") }  
 var dropLocation by remember { mutableStateOf("") }  
 var paymentMode by remember { mutableStateOf("") }  
  
 Column(modifier = Modifier.padding(16.dp)) {  
  
 Spacer(modifier = Modifier.height(30.dp))  
 OutlinedTextField(value = name, onValueChange = { name = it },  
 label = { Text("Full Name") },  
 modifier = Modifier.fillMaxWidth())  
 Spacer(modifier = Modifier.height(8.dp))  
 OutlinedTextField(value = studentId, onValueChange = { studentId = it },  
 label = { Text("Student/Employee ID") },  
 modifier = Modifier.fillMaxWidth())  
 Spacer(modifier = Modifier.height(8.dp))  
 OutlinedTextField(value = contact, onValueChange = { contact = it },  
 label = { Text("Contact Number") }, keyboardOptions = KeyboardOptions(keyboardType = KeyboardType.Phone),  
 modifier = Modifier.fillMaxWidth())  
 Spacer(modifier = Modifier.height(8.dp))  
 OutlinedTextField(value = email, onValueChange = { email = it },  
 label = { Text("Email Address") },  
 modifier = Modifier.fillMaxWidth())  
 Spacer(modifier = Modifier.height(8.dp))  
 OutlinedTextField(value = pickupLocation, onValueChange = { pickupLocation = it },  
 label = { Text("Pickup Location") },  
 modifier = Modifier.fillMaxWidth())  
 Spacer(modifier = Modifier.height(8.dp))  
 OutlinedTextField(value = dropLocation, onValueChange = { dropLocation = it },  
 label = { Text("Drop-off Location") },  
 modifier = Modifier.fillMaxWidth())  
 Spacer(modifier = Modifier.height(8.dp))  
 OutlinedTextField(value = paymentMode, onValueChange = { paymentMode = it },  
 label = { Text("Payment Mode") },  
 modifier = Modifier.fillMaxWidth())  
 Spacer(modifier = Modifier.height(16.dp))  
 Button(onClick = { navController.navigate("home") }) {  
 Text("Submit")  
 }  
 }  
 }  
  
@Preview(showBackground = true)  
@Composable  
fun PreviewBusPassForm() {  
 BusPassForm(navController = rememberNavController())  
}

